

# Chapter 1 So, what exactly is a GIS?

"produced" is misspelled

To most people, what they see as a GIS is in fact just the front-end output layer, such as the maps produced in Google Maps, or the screen on a TomTom navigation device. The reality of it all extends far beyond that; the output layer is very often the end result of many interconnecting programs along with massive amounts of data.

A typical GIS will include desktop applications used to visualize edit and manage the data several different types of backend databases to store the data and in many cases a huge amount of custom written software tools. In fact, GIS is one of the top industries where a programmer can expect to write a very large amount of custom tooling not available from other companies.

Use proper punctuations

We'll explore some of the applications in detail soon, but for now we'll continue with the 100-foot view. A typical GIS processing setup will look something like as in the above image:

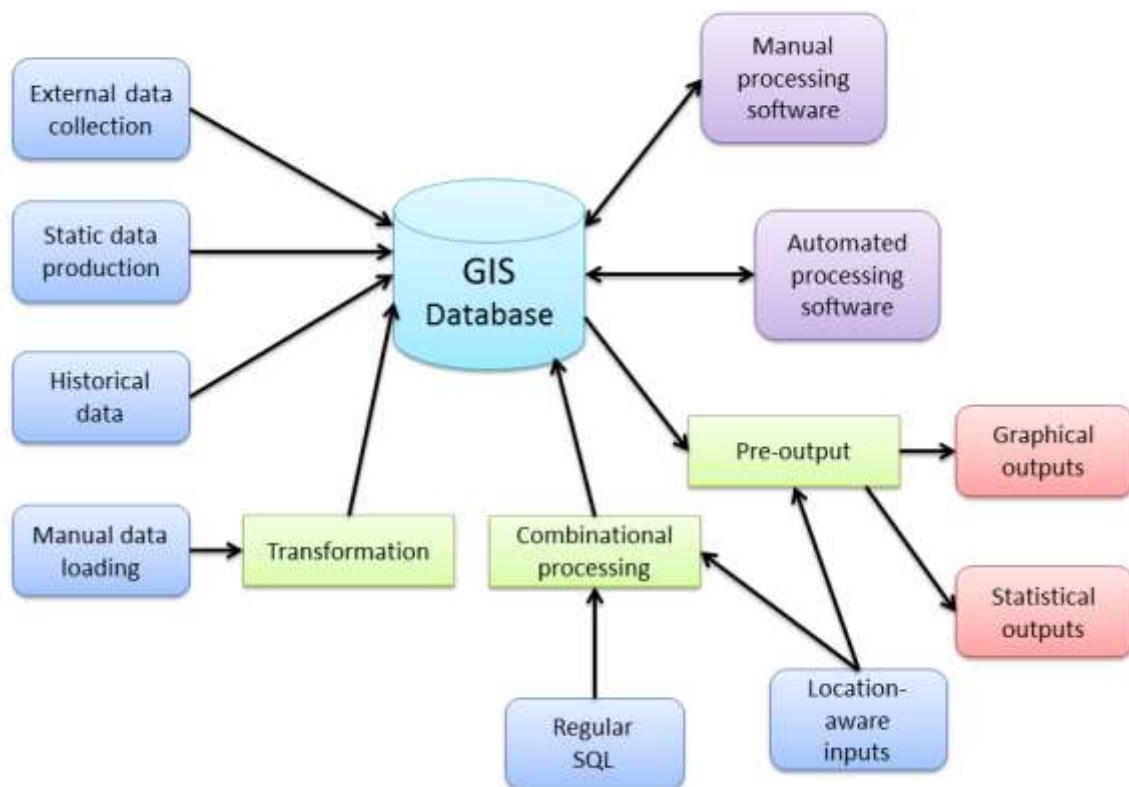



Figure 1: Typical GIS processing setup

As you can see in the diagram, the central part is very often the database itself with a huge number of inputs and processing steps. Finally, the output layers (shown in red) are what people usually associate with being a GIS.


Based on this, we can see that the database is the center of the universe when it comes to GIS.

## The Metadata Tables

All OGC-compliant GIS databases must support two core metadata tables called **geometry\_columns** and **spatial\_ref\_sys**. Most GIS-enabled software will use the existence of these tables to determine if it is talking to a genuine GIS database system. If these tables don't exist, the software will often show an error as shown in the below image. 

A good example of this was with early versions of MySQL where the table names were reserved by the database engine, but did not physically exist as tables. This would cause the MapInfo application to attempt to create the missing tables, but it would receive an error on trying doing so, thus preventing the database from being used correctly by the software.

The **geometry\_columns** table is used to record which table columns in your database contain geospatial data along with their data type, coordinate system, dimensions, and a few other items of related information.

 The **spatial\_ref\_sys** table holds a list of known spatial reference systems, or coordinate systems as they may be better known.

The entries in the **spatial\_ref\_sys** table are indexed by a number known as the EPSG ID. The EPSG, or European Petroleum Survey Group, is a working group of energy suppliers from the oil and gas industry who confronted a common problem that arose when surveying the world's oceans for oil reserves: positioning on a global scale. Some companies used one scale, others used a different scale; some used a global coordinate system, while others used a local one.

The group's solution was to record the differences between each scale and the information required to convert from one scale to another reliably without any loss of precision.

Today, every GIS database that claims to be OGC compliant includes a copy of this table to ensure that data conversions from one system to another are performed with as much accuracy as possible.

We'll cover the actual coordinate systems a little later in the book. For now, all you really need to be aware of is that if the **spatial\_ref\_sys** table does not exist or has no data in it, you will be unable to accurately map or make real-world translations of any data you possess.

Also note that it is possible to save space by removing unnecessary entries from this table. If your data only ever uses two or three different coordinate systems, it's perfectly acceptable to remove the rest of the entries to reduce the size of the table. This can be especially useful when working with mobile devices.

If you only work with data in your own range of values, arguably there can be no data in the **spatial\_ref\_sys** table at all. I would, however, caution you against removing the table entirely. As previously mentioned, most GIS software will look for the presence of this and the **geometry\_columns** table to signify the existence of a GIS-enabled database.